4. Eui

Program Name: Eui.java **Input File: eui.dat**

In her statistics class, Eui came across the birthday paradox, which states that if the distribution of birthdays is uniformly random, and there are just 23 people in a room, there is roughly a 50% chance that at least two people share a birthday. This seems surprising at first glance, since there are 365 days in a year, but the number of pairs of people quadratically grows as the number of people increases.

While checking to see if any of her classmates share birthdays, Eui realized that she doesn't actually know her classmates' exact birth dates. However, she knows their birth months and has a rough estimate of the date itself. Given the information she has, she wants to know the number of possible distinct birth dates.

Input: The first line of input is an integer T (1 <= T <= 50), the number of test cases. Each test case begins with a single integer N (1 <= N <= 200), the number of students in Eui's class. Each of the next N lines contains a date range: the name of the month, the beginning of the date range, a hyphen, then the end of the date range. For example,

```
      January 1 - 3
```

represents the set of dates January 1st, January 2nd, and January 3rd. It is guaranteed that all dates in the input are valid in non-leap years.

Output: For each test case, output the minimum and maximum number of distinct birth dates, formatted with the case number as in the samples.

Sample Input:

```
2
3
January 1 - 3
January 1 - 3
January 1 - 3
5
February 5 - 10
February 3 - 4
March 9 - 12
March 12 - 15
March 9 - 15
```

Sample Output:

```
Case #1: 1 3
Case #2: 3 5
```

Sample Explanation:
In the first sample, there are 3 students, and their birth dates are all between January 1st and January 3rd, inclusive. It is possible they all have distinct birthdays, and it is also possible that they all share a birthday.